End-to-End Learning of Collaborative Loco-Manipulation for Load Transportation using Quadrupeds with Manipulators

Lokesh Kumar*, Sarvesh Sortee*, Titas Bera*

***Abstract*— Collaborative load transportation using quadrupedal robots presents a complex yet highly promising challenge in the field of mobile robotics. Quadrupeds, with their superior mobility over uneven and unstructured terrains, are well-suited for tasks in environments such as disaster zones, off-road logistics, and exploratory missions. However, when tasked with carrying shared payloads, the system becomes significantly more difficult to control due to its underactuated nature where the payload's motion must be regulated indirectly through coordinated whole-body movements of the robots. This demands high levels of synchronization, force distribution, and dynamic balance among agents. To address these challenges, we leverage deep reinforcement learning (DRL) to develop a control policy that enables multiple quadrupedal robots to collaborate in transporting a shared load. The proposed system employs a centralized policy framework, which allows for holistic observation and control during training and deployment. While decentralized policies offer greater scalability and robustness in real-world scenarios, the centralized setup provides a tractable foundation for learning and evaluating fundamental coordination strategies. This paper explores the integration of DRL with centralized control for multi-agent quadrupedal load transportation, highlighting learned behaviors, stability mechanisms, and generalization across payload variations and terrains, laying groundwork for future decentralized extensions.**
Supplementary video is available at `https://youtu.be/VRu4mo3wuZ4`

I. INTRODUCTION

While wheeled robots have seen success in structured indoor logistics, their limitations in off-road environments have driven interest in legged systems. Quadrupedal robots with manipulators, in particular, offer enhanced adaptability, making them ideal for navigating uneven surfaces, stepping over obstacles, and adjusting contact forces dynamically. These features make them promising candidates for cooperative transportation tasks where both navigation and physical interaction with the environment and the payload are required.

A defining aspect of this problem is the need for simultaneous locomotion and manipulation activity. Each quadruped equipped with manipulator arm must coordinate not only its movement across the terrain but also its contribution to the shared control of the payload. This results into a coupled dynamics problem with high-dimensional action and observation spaces, where the success of the task depends on precise, synchronized whole-body motions.

To explore and address this coordination problem, we propose a learning-based approach using deep reinforcement learning (DRL) under a centralized control architecture. Unlike a decentralized policy, which operate on local information, a centralized framework enables the joint optimization of coordination strategies across all agents, facilitating richer learning signals and more coherent global behavior during both the training and inference phases.

Researchers have explored various control methodologies to enhance the coordination and collaboration among multiple robots engaged in load transportation tasks. Caccavale et al. [1] proposed an impedance control approach for cooperative quadrotors equipped with robotic arms, aimed at regulating contact forces during object/environment and manipulator/object interactions. Mellinger et al. [2] addressed

*The authors are with TCS Reseach, India {kumar.lokesh7, sarvesh.sortee, titas.bera}@tcs.com

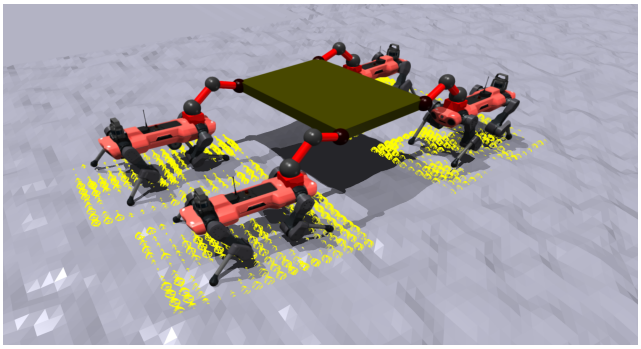


Fig. 1: Quadrupeds with a 4DoF arm carrying a payload

TABLE I: List of Nomenclature

Nomenclature	*Description*
N	Total number of quadrupeds
R_l	Rotation matrix of payload
R_b	Rotation Matrix of quadruped base
V_{lx}, V_{ly}, V_{lz}	Linear velocities of load in x, y and z
ω_{lx}, ω_{ly}, ω_{lz}	Angular velocities of load about x, y and z
V^*_{lx}, V^*_{ly}, ω^*_{lz}	Desired linear and angular velocities of load
V_{bx}, V_{by}, V_{bz}	Linear velocities of quadruped base
ω_{bx}, ω_{by}, ω_{bz}	Angular velocities of quadruped base
F_{lx}, F_{ly}, F_{lz},	Total force on the payload
h_b	Quadruped base height
h^*_b	Quadruped Desired base height
m_l	mass of payload
q_i	Joint positions
$\dot{q}_i$	Joint velocities
$\dot{q}^-_i$	Joint velocities in previous step
τ_i	Joint torques
τ_{i_l}	Joint torque limits
$\mathbb{F}_g$	Feet contact forces with ground
$\mathbb{H}_{terrain}$	Measured terrain heights
$\mathbb{A}$	Actions
$\mathbb{A}^-$	Actions in previous step
$\mathbf{g}$	Gravity vector

the control intricacies of orchestrating multiple quadrotor robots to jointly grasp and transport a payload in three-dimensional space. Nguyen et al. [3] devised the Spherically Connected MultiQuadrotor (SmQ) platform, featuring a rigid frame and multiple quadrotors linked via passive spherical joints, to collectively adjust attitude and thrust force for aerial maneuvers and manipulation. Tagliabue et al. [4] tackled the challenge of moving bulky objects using multiple Micro Aerial Vehicles (MAVs) without direct inter-agent communication, modifying the agents' physical characteristics to ensure stable transportation with resilience. Culbertson et al. [5] introduced a decentralized adaptive controller design enabling a group of agents to manipulate a shared payload in either three-dimensional or two-dimensional space, operating without necessitating inter-agent communication or prior knowledge of agent positions or payload properties. Farivarnejad et al. [6] employed a decentralized sliding mode control strategy for collective payload transport by a team of robots.

Additionally, several other studies [7], [8], [9], [10], have investigated collaborative object transportation, however all these works primarily focus on either aerial or ground vehicles. Although there exists a substantial body of literature concerning multi-robot systems, underscoring their profound importance within the robotics domain, only a limited subset delves into the realm of collaborative loco-manipulation, with even fewer studies specifically addressing this area. Kim et al. [11] presented a layered control framework tailored for real-time trajectory planning and control to achieve robust cooperative locomotion by a pair of holonomically constrained quadrupedal robots. The work interconnected network of reduced-order models grounded in single rigid body (SRB) dynamics, the control architecture consisted of two distinct model predictive control (MPC) algorithms to tackle the optimal control problem concerning the interconnected SRB dynamics. Fawcett et al. [12] devised a model employing principles from behavioral systems theory, tailored for interconnected quadrupeds subjected to holonomic constraints. This model was subsequently integrated into a distributed predictive control framework, enabling each agent to plan its own movements while accounting for the movements of the other agents. Yang et al. [13] took a different approach to tackle the problem of collaborative object transportation by manipulating the payload to specified goal location while avoiding collisions using cables attached to the payload. The incorporation of cables allowed the robotic team to maneuver through confined spaces by altering its inherent dimensions via the toggling of cable tension between slack and taut states. Ji et al. [14] presented a reinforcement learning approach for collaborative quadrupeds, demonstrating the movement of a load by four quadrupeds without manipulators, considering target position and velocity tracking. De Vincenti et al. [15] expanded upon the MPC control techniques established in legged robotics research to encompass collaborative loco-manipulation scenarios. Their approach involved employing a direct multiple shooting method to address the ensuing high-dimensional optimal control challenges pertaining to trajectories of ground reaction forces, manipulation wrenches, and stepping locations, however the robot-self collisions were not taken into consideration.

Contribution

While the aforementioned studies offer valuable insights into the effectiveness of learning-based, distributed control and centralized approaches, the focus is only on multi-agent locomotion and lacks exploration into simultaneous cooperative locomotion and manipulation activities. In [11] and [15], loco-manipulation task planning is considered; however, the real-time applicability of the proposed algorithms and their scalability requires substantial parallelism in computational framework. Additionally the MPC based approach is only validated in a terrain with low degree of surface irregularity. In response to these challenges, we present a deep reinforcement learning (DRL) based approach for achieving collaborative load transportation through loco-manipulation. Our proposed methodology being fairly general empowers multiple quadrupeds equipped with manipulators to cooperatively transport a payload while preserving the desired orientation and user-defined linear and angular velocities of payload in uneven irregular terrain.

The remainder of this paper is organized as follows. Section II - III describes the proposed system architecture and training methodology. Section V reports a series of experiments assessing the performance and generalization capabilities of the learned policy. Finally, we conclude with a discussion of the limitations and potential directions for future work.

II. System Design

We have utilized the ANYmal C quadruped from ANYbotics as the foundational platform for our research, augmenting it with a 4-degree-of-freedom (4DoF) arm atop for loco-manipulation purposes. Each robot comprises 16 actuated joints, with three revolute joints for each leg governing hip adduction/abduction, hip flexion/extension, and knee flexion/extension, alongside four revolute joints dedicated to the manipulator, with three revolute joints at the base and one revolute joint at the elbow. The manipulators are attached to the load using a hinge joint. Figure 1 illustrates the configuration of the robot. To facilitate training across diverse terrains, we adopted environments inspired by Rudin et al.'s work [16]. We train a collaborative loco-manipulation policy over both flat and discretized rough terrains where the height of each cell is sampled uniformly from a range, difficulty level of the terrain is increased gradually as the robots cross there respective levels upto a certain distance.

III. Reinforcement Learning Framework

A. Definitions

Within the framework of reinforcement learning, the decision-making process of an agent is conceptualized through a Markov Decision Process (MDP). An MDP is defined as a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$, encompassing the state set $\mathcal{S}$, the action set $\mathcal{A}$ (i.e., the realm of possible decisions), the transition function $\mathcal{P}$ delineating the impact of selecting an action a_t at time step t on the subsequent state s_{t+1}, often represented as a distribution $\mathcal{P}(s_{t+1}|s_t, a_t)$, and a reward function $\mathcal{R}(s_t, a_t, s_{t+1})$ which quantifies the desirability of an action.

The aim of a reinforcement learning (RL) agent is to enhance the cumulative expected reward throughout a designated time span. In Deep RL, this is frequently achieved by determining a policy $\pi_\theta : [\mathcal{S}, \mathcal{A}] \rightarrow [0, 1]$, where θ denotes a parameter vector. The optimal parameter configuration θ^* is ascertained through an RL algorithm that endeavors to maximize the anticipated reward yielded by trajectories derived from executing the policy π_θ. Mathematically, this optimal parameter set is expressed as:

$$\theta^* =_\theta \sum_{t=1}^{T-1} \mathbb{E}_{\{(s,a)\}_t \sim p_{\pi_\theta}(s_t, a_t)} \mathcal{R}(s_t, a_t, s_{t+1}) \qquad (1)$$

B. Action and Observation Space

1) *Action Space:* As described previously, each robot has 16 actuated joints. We choose to directly learn targets for these $16\times N$ (where N is number of robots used to carryout collaborative loco-manipulation) joints which would then be tracked by actuator level controllers.
2) *Observation Space:* The framework takes in observations as input, as outlined in Table II. Additionally, as blind locomotion is not considered, we sample a grid of measured terrain height surrounding the current position of each quadruped participating in collaborative loco-manipulation. We employ a grid size of 24×12 with a resolution of $10cm$, as indicated in the final row of Table II.

TABLE II: All Observations

Observations	Dimension
V^*_{lx}, V^*_{ly}	2
ω^*_{lz}	1
$R_l^T\,\mathbf{g}$ (Gravity vector expressed in payload frame)	3
$R_b^T\,\mathbf{g}$ (Gravity vector expressed in quadruped frame)	$3\times N$
V_{bx}, V_{by}, V_{bz}	$3\times N$
ω_{bx}, ω_{by}, ω_{bz}	$3\times N$
h_b	$1\times N$
q_i	$16\times N$
$\dot{q}_i$	$16\times N$
$\mathbb{A}^-$	$16\times N$
$\mathbb{H}_{terrain}$	$24\times12\times N$

C. Rewards

We design reward functions aimed at incentivizing robots to maintain specific payload orientation and adhere to commanded *x*-*y* linear and angular velocities. Undesirable movements, such as vertical velocities and angular velocities along the payload's x and y axes, incur penalties.

TABLE III: Reward Formulation

Rewards	Formulation	Scales
linear velocity x	$exp(-\|\|V^*_{lx}-V_{bx}\|\|^2)$	1.0
linear velocity y	$exp(-\|\|V^*_{ly}-V_{by}\|\|^2)$	1.0
Angular velocity z	$exp(-\|\|\omega^*_{lz}-\omega_{bz}\|\|^2)$	1.0
linear velocity z	$\|\|V_{bz}\|\|^2$	-2.0
Angular velocities xy	$\|\|\omega_{bx}+\omega_{by}\|\|^2$	-0.02
Orientation payload	$\|\|\mathbf{g}-R_l^T\,\mathbf{g}\|\|^2$	-2.0
Orientation quadrupeds	$\|\|\mathbf{g}-R_b^T\,\mathbf{g}\|\|^2$	-1.0
Quadruped base height	$\|\|h^*_b-h_b\|\|^2$	-1.0
Torques	$\Sigma_{i=1}^{joints}\ \tau_i$	-0.00002
Acceleration	$\Sigma_{i=1}^{joints}\ \|\|\dot{q}_i^- - \dot{q}_i\|\|^2$	-2.5e-7
Torque limits	$\Sigma_{i=1}^{joints}\ \|\tau_{i_l} - \tau_{i_l}\|$	-0.5
Actions	$\Sigma_{i=1}^{joints}\ \|\|\mathbb{A}^- - \mathbb{A}\|\|^2$	-0.01
Collision		-1.0
Termination		-1.0

We promote smoothness by penalizing both actuator torques and variations in actuator targets. Moreover, collisions between the robot's upper and lower legs and the terrain are penalized, while collisions involving the robot base and payload are deemed terminal, resulting in environment reset. Also to be mentioned the self collisions of the robots had been taken into consideration, thus ensuring that any point of time the arms are in a safe configuration. Drawing inspiration from [16], we integrate a reward component to encourage longer intervals between successive foot contacts with the terrain. The reward formulation has been shown in Table III.

D. Learning Algorithm and Policy Network

The $\mathbb{H}_{terrain}$ information of each quadruped is separately fed to the three CNN layers as shown in Fig. 3 the output received is a latent representation of $\mathbb{H}_{terrain}$ which has a dimension of 64, these latent height representations for each robot are concatenated resulting in a $64\times N$ dimension vector which is then stacked with other observations and fed to a feed forward network with fully connected layers. The network outputs the joint torques for all quadrupeds involved in collaborative load transportation i.e $16\times N$ which are then fed to move the joints in simulation.

TABLE IV: PPO Hyper-parameters and NN shapes

No. of robots ($n = N$)	2048
Batch size ($N \times T$)	2048×24
Mini batch size	2048×6
Number of learning epochs	5
Entropy Coefficient	0.01
Discount Factor	0.99
Advantage Estimator Discount Factor	0.95
Desired KL-Divergence	0.01
CNN kernel size	3×3
Activation	Rectified Linear Unit
CNN input layer	$24\times12\times1$
CNN first layer	$24\times12\times16$
CNN second layer	$12\times6\times32$
Feedforward hidden layers	[512,256,128]
Activation	Exponential Linear Unit
Learning rate	adaptive

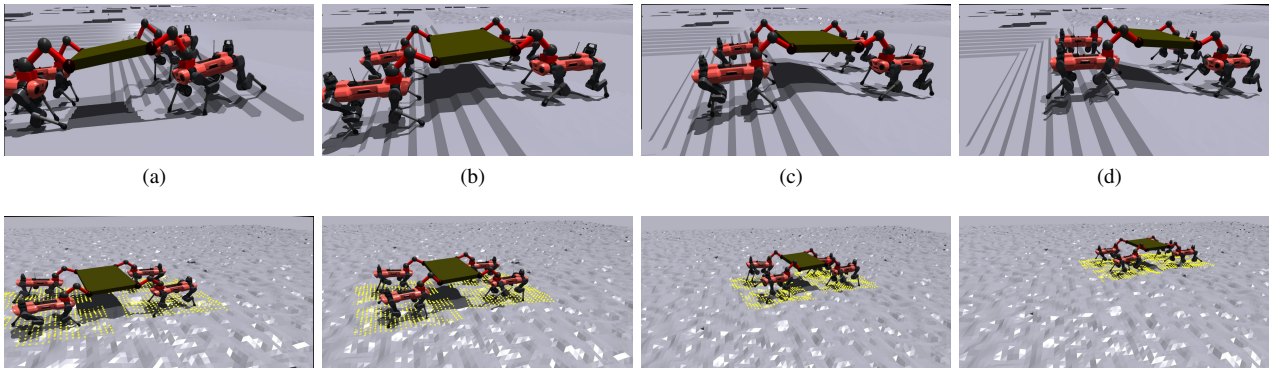


Fig. 2: Sequence of images showing four quadrupedal robots collaboratively transporting a rectangular payload across challenging terrains. The top row illustrates successful negotiation of stair-like structures, while the bottom row demonstrates traversal over rough terrain (yellow dots represent the height samples $\mathbb{H}_{terrain}$ for each quadruped).

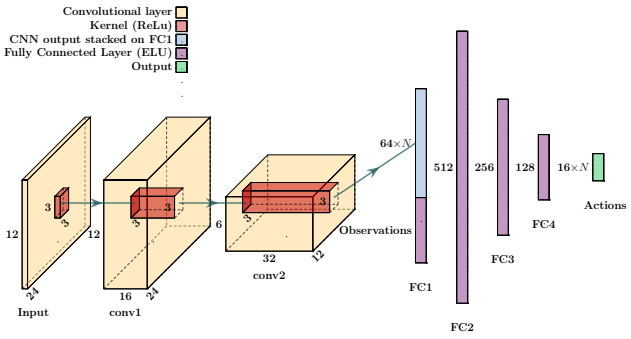


Fig. 3: Network Architecture: Here the CNN layers are shown only once, however there are N CNN's for extracting latent representation of $\mathbb{H}_{terrain}$ for each quadruped.

We employ the Proximal Policy Optimization (PPO) algorithm [17] for learning our control policy over continuous state and action spaces. The main advantages of PPO are that it is computationally cheaper due to its simplicity, requires minimal algorithm hyper-parameter tuning and better sample efficiency. The learning methodology learns two networks, a policy network π_θ and a advantage estimation(critic) network V_ϕ parameterized by θ and ϕ respectively. While, in general, π_θ and V_ϕ may have differing architectures, for simplicity we take that both π_θ and V_ϕ neural networks are fully connected with identical layer sizes. Table IV shows layers shapes for the neural networks and values of PPO hyper-parameters.

E. Training Methodology

We use NVIDIA's Isaac Gym [18], [16] with PhysX as the simulation engine for its ability to simulate thousands of robots in parallel and entirely on the GPU. The policy is trained on different terrains (flat, rough, stairs with maximum step height 7cm) representing moderately harsh but physically realistic obstacles such as curbs, small stairs, and industrial floor discontinuities. For the ANYmal C platform, this step height corresponds to approximately 25% of the robot's shank length (length of the segment below the knee), and about 15% of its ground clearance in stance, introducing substantial foothold and stability challenges, particularly under payload conditions. To improve training stability and generalization, a curriculum learning strategy is applied over both terrain difficulty and commanded payload velocities. Training begins on flat and mildly rough terrains with low velocity commands and progressively advances to rough surfaces and stair obstacles as performance exceeds predefined success thresholds. Likewise, the group is demoted if the performance is unsatisfactory. For payload velocity commands, the curriculum defines the range from which commands are sampled with the maximum of the range increasing with difficulty. The group is promoted if average velocity tracking reward obtained during the training episode exceeds 80% of the maximum and demoted if it is less than 40% of the maximum. For terrains, the curriculum determines the unevenness. The difficulty of each type of terrain increases along the rows in the terrain grid. A group is promoted to next terrain difficulty if its travelled distance is more than 80% of that determined by the command velocity and demoted if it is less than 50%.

A total of 2048 groups for a total of 8192 robots are simulated at once to collect environment interaction data. Each training episode last for a maximum of 30 seconds. At the beginning of the training episode each group is spawned over an appropriate terrain based on its terrain curriculum level, payload command velocities are sampled uniformly for each group according to their level in the command curriculum. Additionally command velocities are resampled again at the 20 second mark in the episode.

To make the learned control policy robust to the variations in the environment, friction coefficient between the robot feet and the terrain is uniformly randomized in the range of $[0.5, 1.25]$ and the payload mass (m_l) is uniformly modified by ± 10 Kg from a nominal value of 15 Kg. Robustness to disturbances is tackled by artificially introducing velocity perturbations to the payload in the range $[0, 1]$ m/s in random

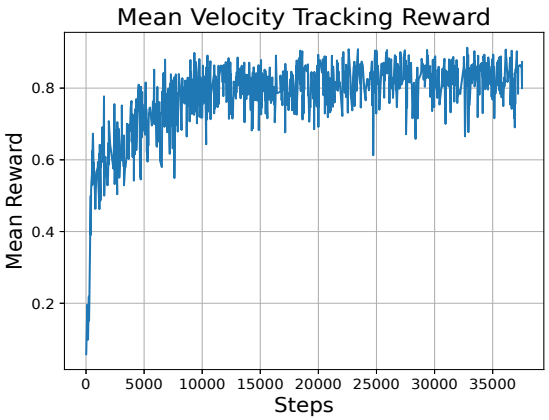


Fig. 4: Mean velocity tracking rewards (maximum value=1).

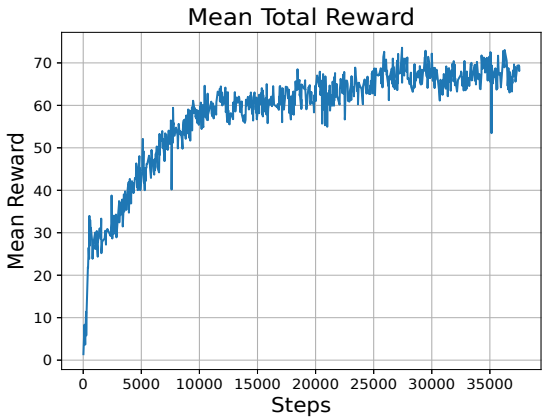


Fig. 6: Mean Total Rewards

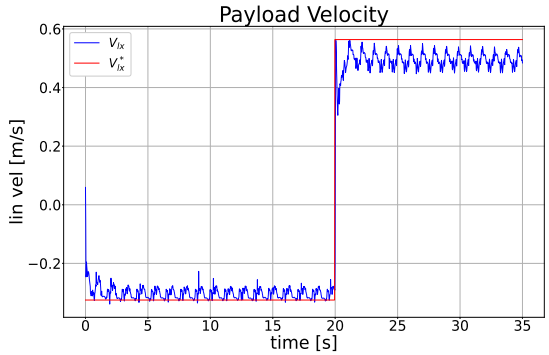


Fig. 5: Payload Velocity V_{lx}

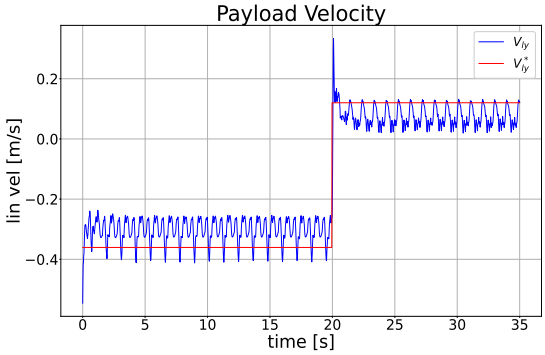


Fig. 7: Payload Velocity V_{ly}

directions at the 15 second mark in the episode.

IV. Results and Discussions

We train a control policy for a group of 4 legged mobile manipulators carrying a rectangular payload. The training was performed on a machine with an Intel Core-i9 processor, 32GB RAM and a 16GB Nvidia 4090 GPU. The average total reward achieved by the learned policy is shown in Fig. 6, whereas Fig. 4 shows the mean of the payload velocity tracking component of the total reward.

We perform experiments with the learned controller to showcase tracking and disturbance rejection capabilities. These experiments are performed with a payload of m_l = 10 kilograms on a discretized terrain of resolution 10cm where the height of each cell is sampled uniformly in $[-0.1, 0.1]$ meter. Total nominal mass of four robots along with the payload is around 255 kilograms.

A. Simulation Experiments

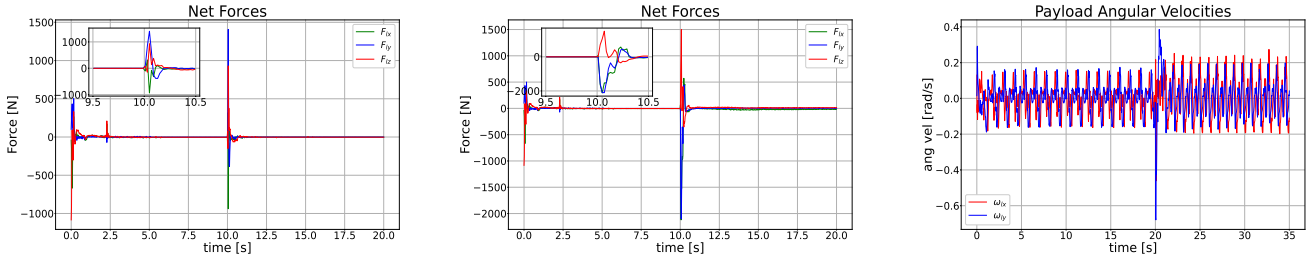
1) Tracking: We assess the tracking performance by commanding a fixed payload linear velocity in the *xy*-plane and zero angular velocity. The desired and achieved components of the payload linear velocities are shown in Figs. 5 and 7. Fig. 8c shows the angular velocities along the x and y axes, it can be observed that the angular velocities are close to zero mark depicting that the load is always maintained flat to the horizontal surface. Note that the commands are changed at 20 seconds.

2) Disturbance Rejection: To evaluate the disturbance rejection and re-balancing capabilities of the robots, we introduce external perturbations to the payload at regular intervals during transportation. Specifically, an impulse-like perturbation is applied every 15 seconds by imparting a velocity to the payload as described in III-E, simulating unexpected disturbances. Figure 8b shows the net forces on the payload when a horizontal perturbation of 2.8 m/s is applied at a $45°$ angle in the plane. Figure 8a illustrates the response to a vertical perturbation of 2 m/s applied in the -z direction. Notably, these perturbations exceed the magnitude of disturbances encountered during training, allowing us to assess the robustness and generalization of the learned policy.

In both scenarios, the desired behavior is to keep the payload stationary. We observe that, immediately following the disturbance, the robots increase the net forces exerted on the payload to counteract the impulse. These forces rapidly decay back toward zero as the system re-stabilizes, effectively rejecting the disturbance.

B. Limitations

Energy efficiency is a critical factor for legged robots operating under payload constraints, although the learned



(a) (b) (c)

Fig. 8: Net forces on the payload under disturbance, and payload angular velocities.

policy successfully adapts to terrain variations, it may not explicitly minimize actuator effort or power usage. Hence incorporating energy-aware objectives, such as minimizing joint torques, electrical power consumption, or cost of transport, could lead to more sustainable locomotion strategies. The curriculum learning strategy prioritizes stability and robustness, which may result in conservative motion patterns under challenging terrains.

V. Conclusion

In this work, we propose a centralized deep reinforcement learning (DRL) policy for collaborative payload transportation using a team of legged mobile manipulators across diverse terrains. The learned policy demonstrates effective tracking of desired payload velocity commands in the horizontal plane and exhibits robust re-balancing and foot repositioning behaviors in response to external disturbances.

However, limitation of the centralized framework is the need for retraining when the number of participating quadrupeds changes. Transitioning to a decentralized or partially decentralized approach could offer improved scalability, robustness, and flexibility in real-world deployment.

Further examinations of gait adaptations, changes in base posture, and load-sharing dynamics across different terrains would provide a more comprehensive understanding of how the robots coordinate under increasing difficulty. Extending the framework towards a sim-to-real transfer with hardware experiments would enable validation of energy usage, actuator stress, and real-world efficiency. Future extensions could introduce objectives that balance robustness with traversal speed, enabling faster payload transport while maintaining safety margins.

References

[1] F. Caccavale, G. Giglio, G. Muscio, and F. Pierri, "Cooperative impedance control for multiple uavs with a robotic arm," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 2366–2371.

[2] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, "Cooperative grasping and transport using multiple quadrotors," in *Distributed Autonomous Robotic Systems: The 10th International Symposium*. Springer, 2013, pp. 545–558.

[3] H.-N. Nguyen, S. Park, J. Park, and D. Lee, "A novel robotic platform for aerial manipulation using quadrotors as rotating thrust generators," *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 353–369, 2018.

[4] A. Tagliabue, M. Kamel, R. Siegwart, and J. Nieto, "Robust collaborative object transportation using multiple mavs," *The International Journal of Robotics Research*, vol. 38, no. 9, pp. 1020–1044, 2019.

[5] P. Culbertson and M. Schwager, "Decentralized adaptive control for collaborative manipulation," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 278–285.

[6] H. Farivarnejad, S. Wilson, and S. Berman, "Decentralized sliding mode control for autonomous collective transport by multi-robot systems," in *2016 IEEE 55th conference on decision and control (CDC)*. IEEE, 2016, pp. 1826–1833.

[7] J. Wehbeh, S. Rahman, and I. Sharf, "Distributed model predictive control for uavs collaborative payload transport," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 11 666–11 672.

[8] H. Yang and D. Lee, "Hierarchical cooperative control framework of multiple quadrotor-manipulator systems," in *2015 ieee international conference on robotics and automation (ICRA)*. IEEE, 2015, pp. 4656–4662.

[9] T. Machado, T. Malheiro, S. Monteiro, W. Erlhagen, and E. Bicho, "Multi-constrained joint transportation tasks by teams of autonomous mobile robots using a dynamical systems approach," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 3111–3117.

[10] M. Campos and V. Kumar, "Guilherme as pereira," *The International Journal of Robotics Research*, vol. 23, no. 7-8, pp. 783–795, 2004.

[11] J. Kim and K. Akbari Hamed, "Cooperative locomotion via supervisory predictive control and distributed nonlinear controllers," *Journal of Dynamic Systems, Measurement, and Control*, vol. 144, no. 3, p. 031005, 2022.

[12] R. T. Fawcett, L. Amanzadeh, J. Kim, A. D. Ames, and K. A. Hamed, "Distributed data-driven predictive control for multi-agent collaborative legged locomotion," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9924–9930.

[13] C. Yang, G. N. Sue, Z. Li, L. Yang, H. Shen, Y. Chi, A. Rai, J. Zeng, and K. Sreenath, "Collaborative navigation and manipulation of a cable-towed load by multiple quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 041–10 048, 2022.

[14] Y. Ji, B. Zhang, and K. Sreenath, "Reinforcement learning for collaborative quadrupedal manipulation of a payload over challenging terrain," in *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2021, pp. 899–904.

[15] F. De Vincenti and S. Coros, "Centralized model predictive control for collaborative loco-manipulation," *Proceedings of Robotics: Science and System XIX*, p. 050, 2023.

[16] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," in *Conference on Robot Learning*. PMLR, 2022, pp. 91–100.

[17] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017. [Online]. Available: https://arxiv.org/abs/1707.06347

[18] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State, "Isaac gym: High performance GPU based physics simulation for robot learning," in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.